



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Patrick Klinecz

Taideteosten kopiointi virtuaalitodellisuussympäristössä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintätekniikan tutkinto-ohjelma

Insinöörityö

15.9.2019

Tekijä Otsikko	Patrick Klinecz Taideteosten kopiointi virtuaalitodellisuusympäristössä
Sivumäärä Aika	34 sivua 15.9.2019
Tutkinto	insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintätekniikka
Ammatillinen pääaine	Software Engineering
Ohjaajat	Lehtori Antti Laiho
<p>Insinööriyön tavoitteena oli luoda prototyyppi järjestelmästä, jossa pelaaja voi virtuaalitodellisuusympäristössä luoda maalauksia ja patsaita sekä verrata niitä mallikappaleeseen. Projektin tilaajana oli Critical Charm Oy.</p> <p>Critical Charm Oy halusi prototyypin mekaniikasta, jonka ympärille voi rakentaa virtuaalitodellisuuspelein. Tarkoituksena oli selvittää, millä tavalla taiteen luonti onnistuu parhaiten ja pystytäänkö kehittämään luotettavasti toimiva vertailumekaniikka sekä maalauksille että patsaille. Prototyyppi kehitettiin Unity-pelimootorilla, ja virtuaalitodellisuusalustana oli Oculus Rift S.</p> <p>Insinööriyön tuloksena saatiin visuaalisesti karkean näköinen, mutta mekaniikoiltaan hyvin toimiva prototyyppi. Pelaaja voi poimia maalausten ja patsaiden paloja, muuttaa niiden väriä ja kokoa sekä rakentaa niistä maalauksia ja patsaita. Pelaaja voi myös verrata valmiita taideteoksia mallikappaleisiin, ja järjestelmä kertoo prosentuaalisesti, kuinka lähelle pelaaja pääsi mallia. Pelintekijä voi luoda uusia malleja helposti 3D-objekteista. Vaikka pelissä ei maalata, vaan maalaukset luodaan mosaiikkimaisesti, aiheuttaa ohjaimien käsien seuranta ongelmia. Tämän takia pelaajan on välillä vaikea sijoittaa palasia oikeisiin kohtiin. Muuten prototyyppi toimii, kuten oli tarkoituskin, ja se on helposti jatkokehitettävissä.</p> <p>Jatkokehitys tapahtunee myöhemmässä vaiheessa, kun Critical Charm Oy:n edellinen projekti on valmistunut. Prototyyppiä voidaan käyttää suoraan projektissa ja sen ympärille on helppo rakentaa kokonainen peli.</p>	
Avainsanat	Virtuaalitodellisuus, pelit, Unity, Oculus

Author Title	Patrick Klinecz Art creation in virtual reality
Number of Pages Date	34 pages 15 September 2019
Degree	Bachelor of Engineering
Degree Programme	Information and Communication Technology
Professional Major	Software Engineering
Instructors	Antti Laiho, Senior Lecturer
<p>The goal of the thesis was to develop a prototype of a system where the player can, in virtual reality, create paintings and statues and compare them to a model. The project was done for Critical Charm Oy.</p> <p>Critical Charm wanted a prototype of a game-mechanic around which to build a game. The main goals were to find out the best way of creating art and a way to create a reliable comparison mechanic for both paintings and statues. The prototype was created using Unity Engine and Oculus Rift S.</p> <p>The result of the thesis was a rough-looking, but mechanically sound prototype. The player can pick up pieces of the paintings and statues, change their colour and size and build paintings and statues with them. The player can also compare the artwork they built against a model artwork and the system will tell how close they player got. The game developer can create new model pieces quickly out of different 3D-objects. However, the hand-tracking causes some issues. Players might have difficulties in placing the pieces where they want. Otherwise the prototype works well and will be easy to build on.</p> <p>Further development will happen later, after Critical Charm Oy is done with their current project. In conclusion, the prototype is completely usable in its current form and it will be relatively straightforward to build a game around it.</p>	
Keywords	Virtual Reality, Unity, Oculus

Sisällys

1	Johdanto	1
2	Virtuaalitodellisuus ja taiteen luonti	2
2.1	Virtuaalitodellisuuden historia	2
2.2	Taiteen luonti virtuaalitodellisuudessa	11
3	Taiteenluontiprototyypin suunnittelu	15
3.1	Projektin kuvaus ja tavoitteet	15
3.2	Haasteet ja alustavat ratkaisut	16
4	Taiteenluontiprototyypin toteutus	18
4.1	Maalausten ja patsaiden luonti	18
4.2	Maalausten ja patsaiden vertailu malliin	21
4.3	Ongelmat ja ratkaisut	27
5	Tulokset	28
5.1	Projektin saavutukset ja analyysi	28
5.2	Jatkokehitys valmiiksi tuotteeksi	29
6	Yhteenveto	31
	Lähteet	32

1 Johdanto

Virtuaalitodellisuus on pitkäikäinen teknologia, mutta vasta viime vuosina tietokoneiden suorituskyky on noussut sellaiselle tasolle, että virtuaalitodellisuusteknologiaa voidaan järkevästi kehittää ja käyttää. Viime vuosina on myös julkaistu useampia taidesovelluksia virtuaalitodellisuusalustoille. Nämä käsittävät erilaisia 3D-piirtotyökaluja ja 3D-veistotyökaluja.

Insinööritöön tarkoituksena oli tuottaa prototyyppi virtuaalitodellisuuspeliä varten. Prototyypin tavoitteena oli kehittää järjestelmät, joilla pelaaja voi luoda virtuaalitodellisuusympäristössä maalauksia ja patsaita, sekä vertailla näitä luomuksia mallikappaleeseen. Maalausten luominen tapahtuu mosaiikkityylisesti. Pelaaja voi värjätä palasia ja muuttaa niiden kokoa ja tämän jälkeen liimata ne kankaalle. Patsaiden kokoaminen tapahtuu samalla tavalla. Lopuksi algoritmi vertaa pelaajan taideteosta mallikappaleeseen ja kertoo tuloksen.

Projektiin kuului molempien järjestelmien suunnittelu ja toteutus. Työn toteutukseen käytettiin Unity-kehitysalustaa ja Microsoft Visual Studiota. Virtuaalitodellisuusalustana toimi Oculus Rift S.

Toimeksianto saatiin Critical Charm Oy:ltä. Critical Charm on suomalainen pelialan yritys, joka kehittää virtuaalitodellisuuspelejä. Yrityksen seuraavaan projektiin kuuluu keskeisenä osana taideteosten kopiointi. Projektin tarkoitus oli tutkia ja kokeilla, kuinka tällaisen järjestelmän saisi toteutettua Unity-pelimootorilla ja virtuaalitodellisuudessa.

2 Virtuaalitodellisuus ja taiteen luonti

2.1 Virtuaalitodellisuuden historia

Virtuaalitodellisuus tarkoittaa Kielitoimiston sanakirjan mukaan ”tietokonesimulaation tuottamien aistimusten avulla luotu keinotekoinen ympäristö, keino-, lume-, tekotodellisuus” [1]. Tämä määritelmä on häilyvä ja sanalla ”virtuaalitodellisuus” on tarkoitettu usein eri asioita. Tästä syystä sen historian alkupistettä on vaikea paikantaa. Nykyään termi on kuitenkin vakiintunut tarkoittamaan erilaisia virtuaalitodellisuuslaseja. Nykyisenlaiset virtuaalilasit ovat olleet olemassa huomattavasti vähemmän aikaa kuin ensimmäiset yritykset virtuaalitodellisuusteknologian luomisessa.

Ensimmäinen varsinainen virtuaalitodellisuuslaite patentoitiin vuonna 1962 Morton Heiligen toimesta. Laite oli kuvassa 1 näkyvä Sensorama, ja sen tarkoitus oli tarjota käyttä-

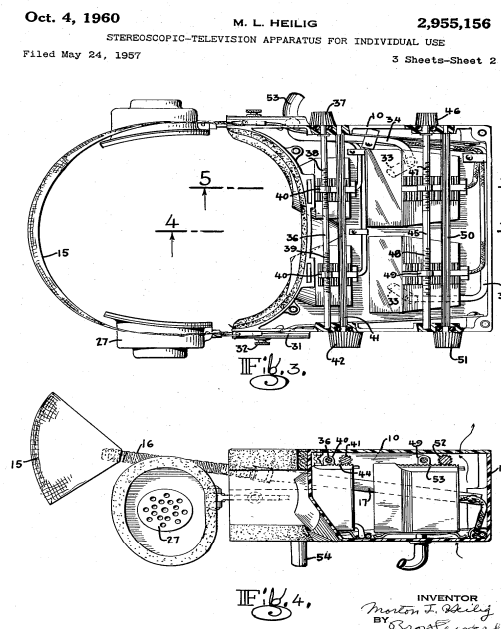


Kuva 1. Morton Heiligen Sensorama [2].

jälleen immersiivinen kokemus, johon kuului 3D-videokuvaa toistettuna stereoskooppisen värinäytön kautta, tuulettimet, hajua erittävät osat, stereoäänilaitteisto sekä liikkuva istuinosa. Yhdistelemällä näitä kaikkia luotiin aikaansa nähden vakuuttava illuusio katsojalle. [3.]

Koska 3D-elokuvia ei ollut tarjolla siihen aikaan, Heilig rakensi myös oman kameransa, jolla hän kuvasi materiaalia näytettäväksi Sensoramalla. Näitä "virtuaalikokemuksia" valmistui kuitenkin vain viisi. Neljässä katsoja ajoi erilaisilla kulkuvälineillä (helikopteri, mikroauto, polkupyörä ja moottoripyörä) ja viidennessä katsoja seurasi newyorkilaisen vatsatanssijan esitystä. Nämä kaikki olivat passiivisia kokemuksia, joissa katsoja oli todellakin vain katsoja. Eri tuoksut ja äänet laukaistiin oikeaan aikaan kokemuksen rikastamiseksi, mutta katsoja ei kuitenkaan itse voinut mitenkään vaikuttaa materiaaliin vaan kyseessä oli normaalia immersiivisempi elokuva. [2.]

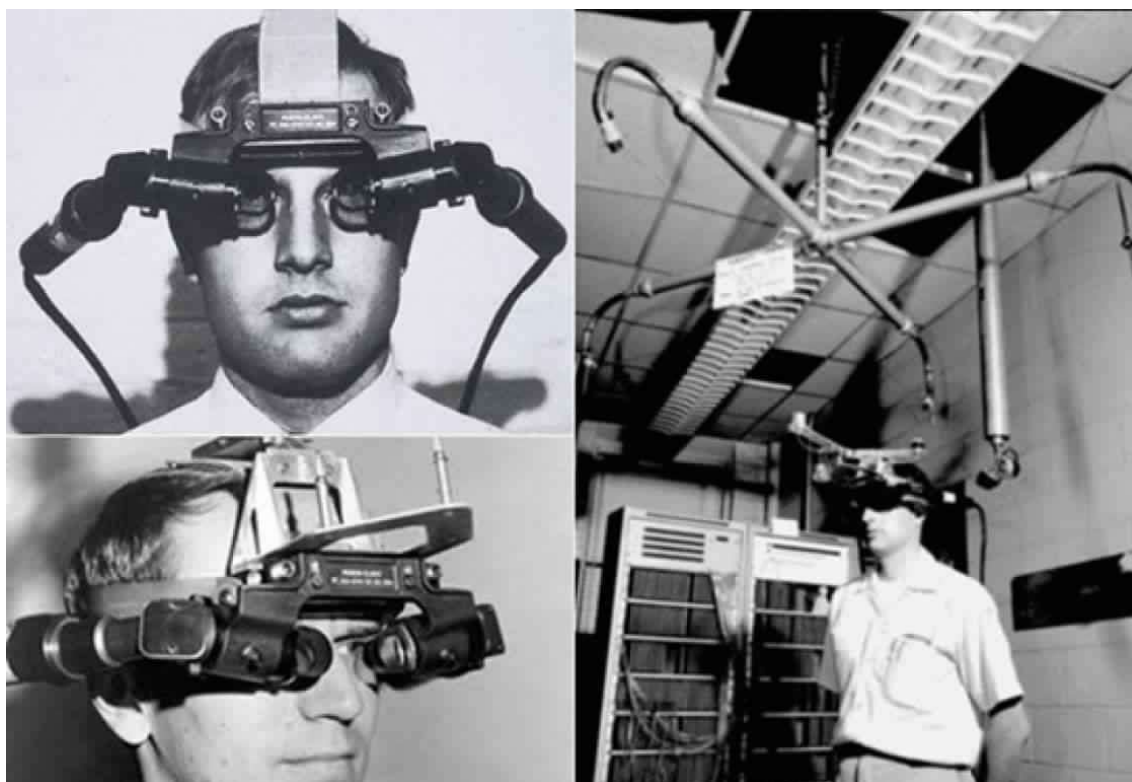
Sensorama oli ison juoma-automaatin kokoinen, mutta Heilig patentoi myös ensimmäisen päälle puettavan virtuaalitodellisuusmaskin, kuvassa 2 olevan Telesphere Maskin. Laite koostui linseistä, kahdesta pienestä kuvaputkesta, kuulokkeista sekä ilmasuuttimista. [4.]



Kuva 2. Heiligin Telesphere [4].

Heilig vaikutti kuitenkin olevan aikaansa edellä. Elokuvastudiot eivät kiinnostuneet projekteista, vaikka pelihalli-tyyppinen 25 sentin kolikoilla toimiva malli tuottikin hyvin rahaa Universal Studiosilla. Myöskään opetusinstituutiot, joille Telesphere Maskia markkinoitiin opetusvälineenä, eivät kiinnostuneet asiasta. Ilman rahoitusta tuotteiden massatuotanto oli mahdotonta, ja Heilig joutui siirtymään muiden projektien pariin. [2.]

Seuraava kehitys alalla nähtiin, kun Ivan Sutherland ja hänen oppilaansa Bob Sproull kehittivät omat virtuaalitodellisuuslasinsa Utahin yliopistossa vuonna 1968. Projekti oli Sword of Damocles, joka näkyy kuvassa 3. Käyttäjä laittoi lasit päähänsä, ja laite piirsi rautalankamallin virtuaalisesta huoneesta. Iso osa laitteistosta oli kiinnitetty kattoon käyttäjän yläpuolelle, joten liikkuminen lasien kanssa oli hyvin rajattua. Laitteen immersiiivisyys ja interaktiivisuus jäivät hyvin vajavaisiksi, koska laite pystyi näyttämään vain edellä mainittuja rautalankamalleja ja koska käyttäjä pystyi vaikuttamaan virtuaaliseen maailmaan vain päättään liikuttamalla. Laite jäikin lähinnä konseptitasolle ja oli vain osoitus siitä, että virtuaalitodellisuus on yleensä mahdollista toteuttaa. [5.]



Kuva 3. Sword of Damocles [6].

Ala kehittyi 1970-luvulla hitaasti ja suurimmaksi osaksi ei-kaupallisten tahojen toimesta. NASA kehitti omia laitteitaan 1980-luvun taitteessa, ja samoihin aikoihin Yhdysvaltain ilmavoimat kehitti omaa lentosimulaattoria. Perusajatus oli sama kuin Heiligen Telespheressä. Heiligen mielestä oli tärkeää, että henkilöitä voitaisiin opettaa toimimaan oikein vaaratilanteissa, mutta ilman tarvetta altistaa näitä henkilöitä kyseisille vaaratilanteille.

Kaupallisia tuotteita ei kuitenkaan nähty pitkään aikaan. Atari perusti oman virtuaalitodellisuuslaboratorion tutkimaan, kuinka virtuaalitodellisuutta voitaisiin hyödyntää videopeleissä. Tämä tapahtui kuitenkin vuonna 1982 ja jo seuraavana vuonna iski vuoden 1983 videopelialan romahdus, jonka seurauksena Atari oli pakotettu sulkemaan monia osastojaan, mukaan lukien tämän laboratorion. [7.]

Vasta 1990-luvulla virtuaalitodellisuus alkoi todella kehittyä kaupallisesti. Esimerkiksi Computer Gaming World ennusti vuonna 1992, että kuluttajahintaisia VR-laitteita olisi saatavilla jo vuonna 1994 [8]. Virtuality Group toi omat tuotteensa markkinoille ensimmäistä kertaa vuonna 1991. Lasit toimivat Amigan tietokoneilla ja mukana tulivat lasit sekä ohjaimet. Tuote oli kuitenkin liian kallis myytäväksi suoraan kuluttajille, joten näitä pääsi käyttämään lähinnä pelihalleissa.

Sega yritti myös markkinoille kuvan 4 Sega VR -laitteistollaan, josta oli tarkoitus valmistaa sekä pelihalliversio että kotikäyttöön tarkoitettu versio. Loppujen lopuksi kotiversiota ei ikinä saatu valmiiksi ja pelihalliversiokin jäi hyvin rajatuksi julkaisuksi. Sega väitti jul-



Kuva 4. Sega VR:n konseptikuva [9].

kaisun peruutuksen johtuneen liian realistisesta VR-efektistä, jonka takia pelaajat saataisivat vahingoittaa itseään uppoutuessaan peleihin. Ottaen huomioon ajan rajoitukset suorituskyyvyssä todennäköisempi syy löytyy laitteen aiheuttamasta pahoinvoinnista ja päänsärystä. [9.]

Ensimmäinen suoraan kuluttajille myyty VR-pelikonsoli olikin Nintendo Virtual Boy, joka tuli markkinoille vuonna 1995. Kuvassa 5 oleva surullisenkuuluisa laite laitettiin pöydälle telineen varaan, koska se oli liian iso ja raskas pidettäväksi pään ympärille puettuna. Laite toisti ainoastaan kaksiväristä kuvaa (musta ja punainen) ja tämän lisäksi pelien grafiikat olivat käytännössä rautalankamalleja. [10.] Virtual Boy'n tuotanto lopetettiin jo seuraavana vuonna ja konsolille ehti ilmestyä vain 22 peliä. Sitä myytiin vain noin 770 000 kappaletta Japanissa ja Yhdysvalloissa. Euroopassa konsolia ei edes ikinä julkaistu. Samaan aikaan markkinoilla ollut Nintendo Entertainment System myi maailmanlaajuisesti noin 49 miljoonaa kappaletta [11].



Kuva 5. Nintendo Virtual Boy [12].

1990-luvun aikana julkaistiin useita VR-laseja, jotka toimivat PC:llä, mutta yksikään niistä ei onnistunut myymään hyvin. Kallis hinta, grafiikan heikko laatu ja pahoinvointi eivät auttaneet myyntiä. Kuvassa 6 oleva Sonyn Glasstron, Virtual I/O:n iGlasses, Forten VFX1 ja Victormaxxin Cybermaxx kaikki yrittivät vallata markkina-alaa siinä kuitenkin onnistumatta. Kuten 1970-luvulla kävi, kiinnostus teknologiaan hiipui, kun menestystarinat jäivät tapahtumatta eikä kaupallisia tuotteita nähtykään vuosikausiin.



Kuva 6. Sony Glasstron [13].

Pitkän hiljaisuuden jälkeen asiat kuitenkin muuttuivat vuonna 2010, kun Palmer Lucky kehitti ensimmäisen prototyypin Oculus Riftistä. Kuvan 7 Rift oli ensimmäinen kuluttajatasen laite, joka lupasi tarjota 90 asteen katselukulman käyttäjälle. John Carmack, tunnettu peliohjelmoija ja suunnittelija id Softwarelta, kiinnostui projektista ja esitteli sen E3-tapahtumassa vuonna 2012. Samana vuonna Lucky loi Kickstarter-sivustolle rahoituskampanjan tarkoituksenaan kerätä rahaa ensimmäisen version tuotantoon. Kampanja

oli menestys ja tuotti 2,5 miljoonaa dollaria. Alle kaksi vuotta myöhemmin, maaliskuussa 2014, Facebook osti yrityksen 2 miljardilla dollarilla. [14.]



Kuva 7. Alkuperäinen Oculus Development Kit [14].

Tämän jälkeen yritys on tuottanut ja julkaissut useammat VR-lasit. Ensimmäinen kuluttajaversio, Oculus Rift, julkaistiin vuonna 2016 ja sen seuraaja, Oculus Rift S, julkaistiin toukokuussa 2019.

Näiden lisäksi yritys on julkaissut kaksi johdotonta mallia, Oculus Go:n (2018) ja Oculus Questin (2019). Molemmissa lasit itsessään sisältävät prosessorin ja kaiken muun tarvittavan ohjelmien ajamiseen. Lasit liitetään tietokoneeseen vain akun ja ohjelmien latausta varten. Kompromissina on tietenkin alhaisempi suorituskky kuin tietokoneeseen liitettävillä malleilla. Esimerkiksi Oculus Quest käyttää Qualcommin Snapdragon 835 -järjestelmäpiiriä. Samaa piiriä käyttävät useat älypuhelimet, mm. Samsung Galaxy S8, Google Pixel 2 ja Sony Xperia XZ Premium [15]. Tämän takia Oculus Questilla on täysin oma kauppapaikkansa, joka sisältää vain pelejä ja sovelluksia, jotka toimivat laitteella. Iso osa peleistä vaatii optimointia, että Quest kykenee ajamaan niitä, koska Snapdragon

835:n tehot eivät ole lähimaillakaan uuden, tai edes vähän vanhemman, pelitietokoneen tehoja.

Oculukseen kohdistuneen kiinnostuksen myötä myös muiden yritysten kiinnostus alaa kohtaan alkoi kasvaa jälleen. Oculuksen Kickstarter-kampanjan jälkeen onkin julkaistu monia erilaisia VR-laitteita useiden yritysten toimesta.

Google julkaisi oman Cardboard-mallinsa vuonna 2014. Cardboard on todella aloitustason tuote. Nimensä mukaisesti se on tehty pahvista ja sisältää vain tarvittavat linssit. Käyttäjä kiinnittää oman älypuhelimensa pahviseen ulkokuoreen ja katsoo sisältöä Cardboardin linssien läpi. Yksinkertaista rakennetta paikkasi alhainen hinta. Muiden lasien maksaessa 500 dollarista yli tuhanteen, Cardboardin hinta on noin 20 dollaria. Google on sittemmin julkaissut myös muita VR-laseja, mutta kaikkia malleja yhdistää se, että niiden mukana tulee vain linssit, ja käyttäjä tarvitsee älypuhelimien sisällön toistamiseen. [16.]

Kiinalainen elektroniikka-alan yritys HTC ja Steam-kauppapaikasta tunnettu Valve Corporation julkistivat omat yhteistyöllä kehitetyt lasinsa vuonna 2015 ja HTC Vive tuli myyntiin vuonna 2016, vain muutama kuukausi Oculus Riftin jälkeen. Huhtikuussa 2019 Valve julkisti uudet lasinsa, kuvassa 8 näkyvän Valve Indexin, joka tuli myyntiin kesäkuussa 2019.



Kuva 8. Valve Index

Myös Sony lähti uudestaan mukaan VR-kilpailuun. Sen edellä mainittu, vuonna 1997 myyntiin tullut Glasstron, ei menestynyt halutulla tavalla, mutta aika oli kypsä uudelle yritykselle. Vuonna 2016 Sony julkisti kuvassa 9 näkyvän PlayStation VR -laitteistonsa. Lasit liitetään nimensä mukaisesti Sonyn PlayStation 4 -konsoliin, ja pelaaja voi ohjata peliä joko tavallisella PlayStation 4 -ohjaimella tai erikseen ostettavilla Move-ohjaimilla.



Kuva 9. PlayStation VR

Edellä mainittujen lisäksi monilla yrityksillä on omat virtuaalilasit joko myynnissä tai kehityksessä. Näistä mainittakoon Microsoftin Hololens, Samsungin Gear VR, Dell Visor ja Asus HC102.

Yrityksillä on paljon työtä tehtävänä, että VR-lasit voivat murtautua valtavirtaan kunnolla. Tällä hetkellä isoimpina ongelmina ovat laadukkaan sisällön puute sekä laitteistojen hintataso. Vaikka monet lasit, kuten Oculus Rift S, ovat jo suhteellisen halpoja, lasit yksinään eivät riitä.

Lisäksi tarvitaan suorituskykyinen tietokone, jolloin budjetti karkaa helposti jo yli 2000 euron. Mutta kaiken kaikkiaan vaikuttaa siltä, että virtuaalitodellisuus on vihdoin ja viimein lyönyt läpi siihen pisteeseen asti, että samanlaista kiinnostuksen hiipumista ei tapahdu kuten 90-luvun buumin jälkeen.

2.2 Taiteen luonti virtuaalitodellisuudessa

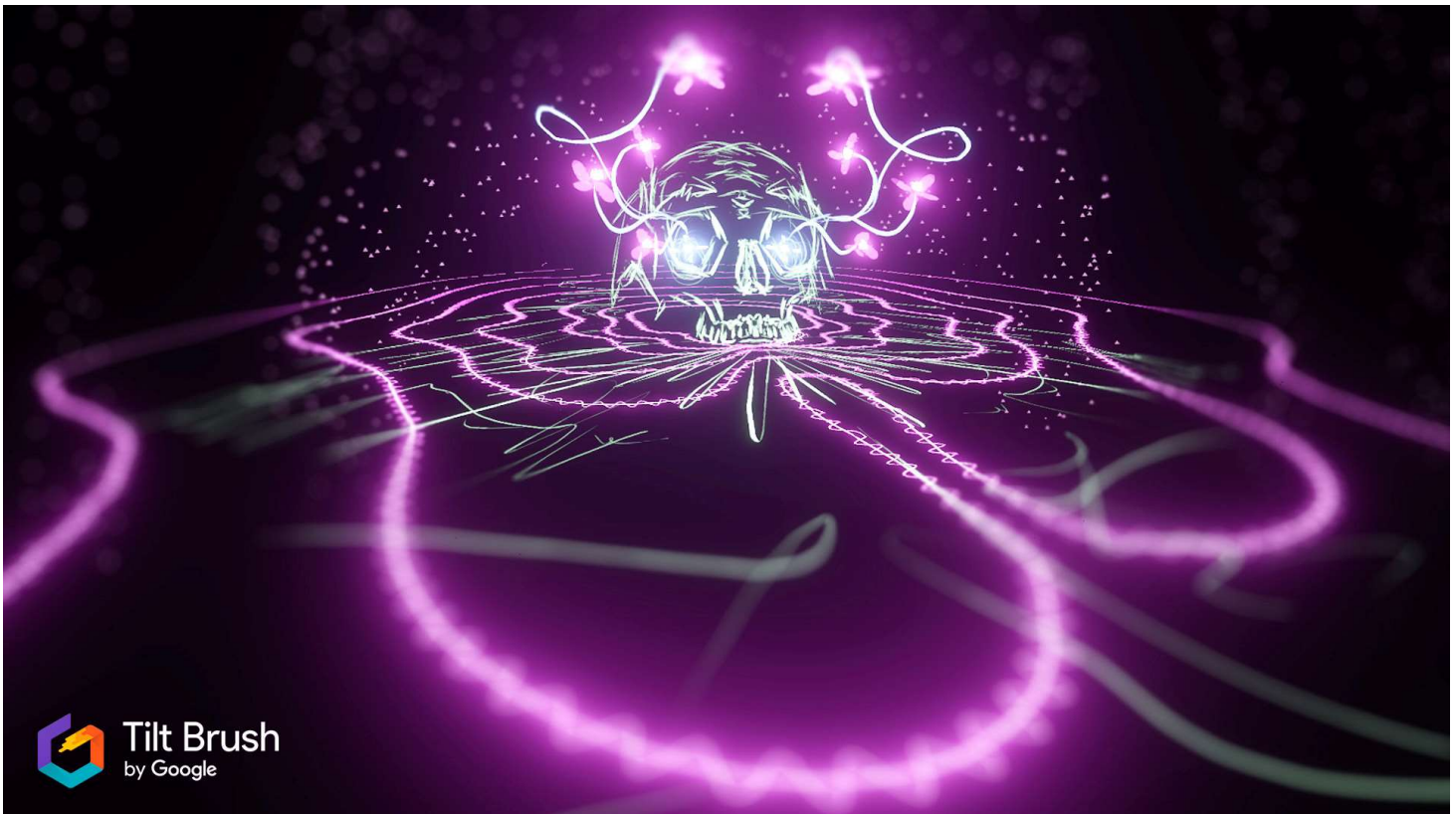
Virtuaalitodellisuus nähdään usein peliteknologiana ja sitä se ensisijaisesti onkin ollut jo pitkään. Virtuaalitodellisuudelle on kuitenkin monia muitakin käyttötarkoituksia [17]. Monet niistä eivät vain ole tarpeeksi kiinnostavia tai vakuuttavia myyntipuheissa, joten ne jätetään huomiotta valtavirtamediassa. Virtuaalitodellisuutta ja lisättyä todellisuutta (AR) käytetään kuitenkin monilla teollisuuden aloilla jatkuvasti enemmän ja enemmän. Esimerkiksi tuoteprototyypin 3D-mallin esittely on paljon vaikuttavampi virtuaalitodellisuusympäristössä ja lisätyn todellisuuden laseja suunnitellaan käytettäväksi esimerkiksi automekaniikoille ja kirurgeille. Samoin arkkitehtuurisuunnitelmat on paljon helpompi ”tuoda eloon” virtuaalitodellisuusteknologialla, kun potentiaalinen asiakas voi päästä suunnitelmaan sisään liikkumaan sen sijaan, että mallia vain katseltaisiin ruudulta.

Pelien ja teollisuussovellusten lisäksi on vielä kolmas tuotealue, kaupalliset sovellukset. Näitä on vielä olemassa vähemmän kuin pelejä, mutta lisää kehitetään jatkuvasti. Esimerkiksi virtuaalimatkailu on uusi alue, joka on vahvassa nousussa [18]. Koko matka voi tapahtua virtuaalisesti, eli käyttäjä ”vieraillee” eri kohteissa virtuaalitodellisuudessa. Jos henkilö haluaa kuitenkin oikeasti lähteä johonkin, virtuaalitodellisuutta voidaan hyödyntää esimerkiksi hotellin toimesta. Asiakas voi kotonaan tutustua virtuaalitodellisuudessa hotellihuoneeseen ja näin helpommin valita itselleen sopivan hotellin ja huoneen [19].

Eräs kaupallisten sovellusten ala on erilaiset taidesovellukset. Niissä käyttäjä voi eri tavoilla luoda piirustuksia, maalauksia, 3D-veistoksia ja malleja ja 3D-piirrustuksia. Tunnetuimpia näistä ovat Googlen Tilt Brush, Facebookin Quill, Oculus Medium, Gravity Sketch ja CoolPaint VR, joka listan ainoana on saatavilla PlayStation VR:lle.

Tilt Brush

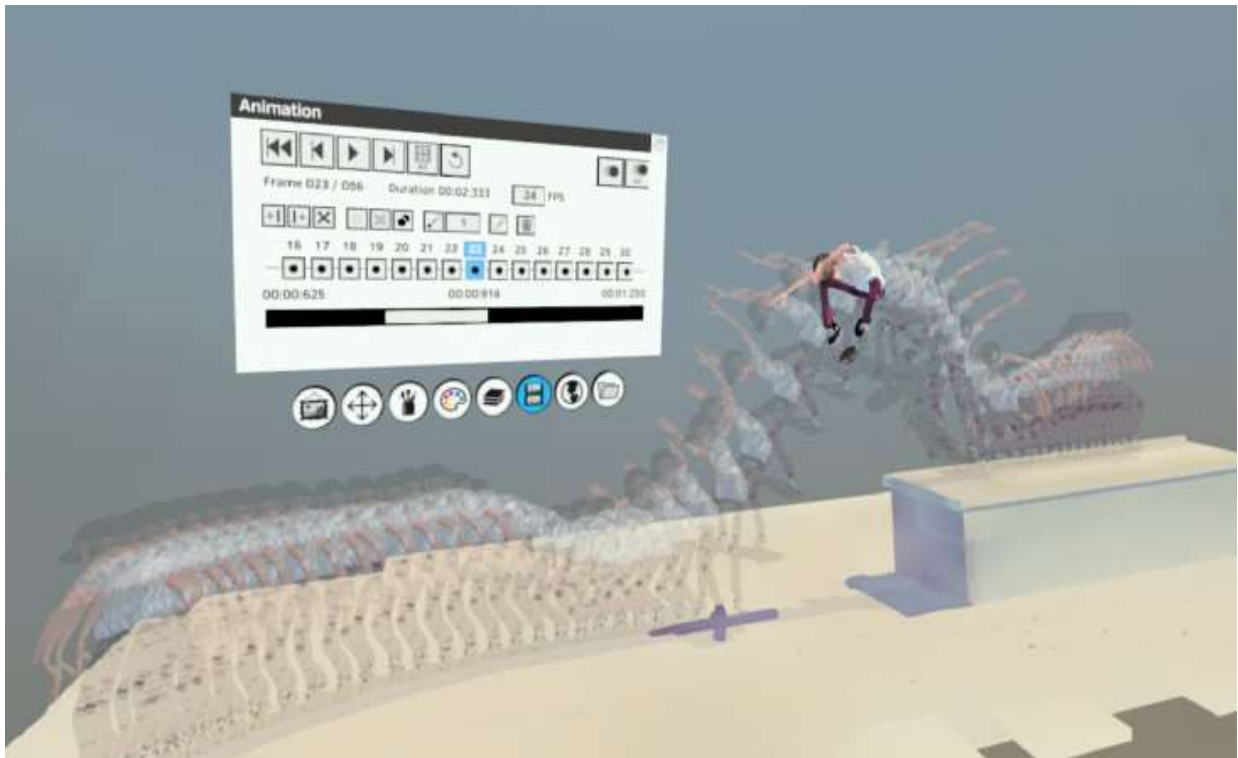
Googlen Tilt Brush on maksullinen sovellus, joka toimii HTC Vivellä, Oculus Riftillä ja Questilla, Windows Mixed Reality -laseilla sekä Valve Indexillä. Se on saatavilla Valven Steam-kauppapaikalta, jossa sen arvosteluista 93 % on positiivisia [20]. Se on myös voittanut roppakaupalla palkintoja ja saanut ylistystä käytännössä kaikista lähteistä. Tilt Brush keskittyy kokonaan kolmiulotteiseen piirtämiseen, joten esimerkiksi 3D-mallien veistämiseksi ei ole tukea, mutta se, mitä sovelluksessa on, on tehty erittäin hyvin. Tilt Brush sisältää erilaisia ”efektimaaleja”. Esimerkiksi musteelle, savulle ja lumelle löytyy omat efektinsä, joita käyttäen voi maalata. Kuvassa 10 on käytetty erilaisia efektejä, kuten tussimaista efektiä kallon piirtämiseen ja valoa hohtavaa efektiä pinkkeihin viivoihin. Tilt Brush sisältää myös helpon työkalun, jolla voi siirtää teokset Unity-pelimoottoriin.



Kuva 10. Esimerkki siitä mitä Tilt Brushilla voi saada aikaan [20].

Facebook Quill

Quill on ilmainen sovellus Oculuksen VR-laseille, joka sisältää hyvin samankaltaiset työkalut kuin Tilt Brush, mutta lisää siihen päälle myös muuta. Quill sisältää täyden tuen animaatioiden luomiselle ja äänen lisäämiselle. Artisti voi työskennellä muissa ohjelmissa luoduilla 3D-malleilla tai luoda omansa suoraan Quillia käyttäen. Käytännössä artisti voi luoda kokonaisen kohtauksen alustavista luonnoksista aina viimeisiin animaatioviilauksiin asti yhdessä virtuaalitodellisuussovelluksessa. Kuvassa 11 näkyy Quillin animaatiotyökalu, jolla animaattori voi määritellä 3D-mallin liikkeen.



Kuva 11. Quillin animaationäkymä [21].

Oculus Medium

Oculus Medium, kuten nimestä saattaa päätellä, on vain Oculuksen laseille saatava taidesovellus, joka keskittyy kolmiulotteisten patsaiden ja veistosten luomiseen ja maalamiseen. Toisin kuin edeltävät sovellukset se ei varsinaisesti tue maalaamista tai piirtämistä.

Gravity Sketch

Gravity Sketch on taidesoftware, joka on saatavilla Oculusille, HTC Viveille ja Windows Mixed Reality -laseille. Se on sovellus, joka tähtää selkeästi enemmän ammattilaismarkkinoille kuin kuluttajamarkkinoille. Käyttötapauksina se listaakin autosuunnittelun, tuotesuunnittelun ja konseptitaiteen. Se kuitenkin sisältää käytännössä kaikki samat asiat, mitä esimerkiksi Facebook Quill tarjoaa. Ainoana lisänä on sen oma Landing Pad -niminen pilvipalvelu, jonka avulla artisti voi siirtää teoksensa suoraan pilveen.

CoolPaint VR

Kuten aikaisemmin on todettu, CoolPaint VR on sovelluksista ainoa, joka on saatavilla PlayStation VR:lle, ja se on saatavilla vain ja ainoastaan tälle alustalle. Verrattuna muihin listattuihin sovelluksiin, se on hyvinkin samanlainen. Hinta on sama kuin Tilt Brushilla ja sillä pystyy tekemään suurimman osan asioista, joita Tilt Brushilla. Kritiikkiä se on saanut siitä, että aivan kaikkeen samaan se ei kykene, mutta tämä johtuu vahvasti PlayStationin rajallisesta suorituskyvystä tietokoneisiin nähden. Pushsquaren mukaan sovellus on riittävän hyvä vaihtoehto Tilt Brushille PlayStationilla, mutta efektien ja objektien määrä jäi vajaaksi [22].

Taulukko 1. Taulukossa perustiedot kaikista esitellyistä ohjelmista

Nimi	Hinta	3D-piirtäminen	3D-mallit	Animointi
Tilt Brush	19,90€	Kyllä	Ei	Ei
Quill	Ilmainen	Kyllä	Kyllä	Kyllä
Medium	0/29,99€	Ei	Kyllä	Ei
Gravity Sketch	27,99€	Kyllä	Kyllä	Ei
CoolPaint VR	19,95€	Kyllä	Ei	Ei

Taulukkoon 1 on kerätty perustiedot kaikista esitellyistä ohjelmista. Oculus antaa Mediumin ilmaiseksi kaikille Oculus Riftin ostajille, mutta muuten se maksaa 29,99€.

3 Taiteenluontiprototyypin suunnittelu

Projektin idea tuli Critical Charm Oy:ltä. Yrityksellä on suunnitelmana toteuttaa peli, johon kuuluu olennaisena osana taideteosten luominen ja näiden vertaaminen malliin. Yritys oli ehtinyt suunnittelemaan peliä jo jonkun verran, mutta tarkennettu suunnitelma siitä, mitä taideteosten luominen ja vertailu käytännössä sisältää, tehtiin vasta insinööri-työn aiheen varmistuttua.

3.1 Projektin kuvaus ja tavoitteet

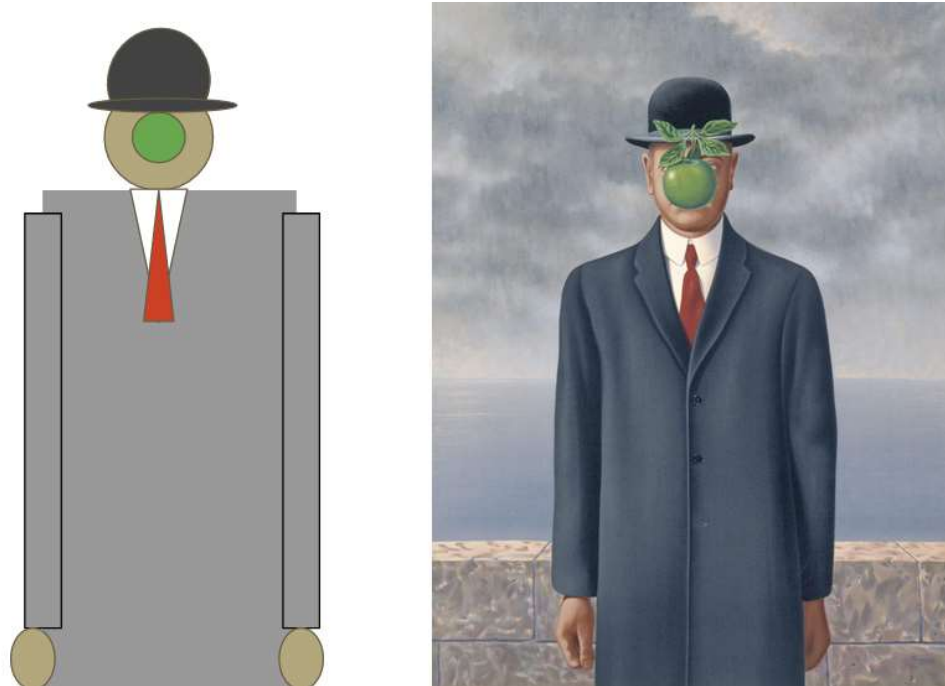
Projektin tavoitteena oli tuottaa prototyyppi, jossa pelaaja voi luoda maalauksia ja patsaita ja sen jälkeen verrata niitä malliin. Prototyypin oli tarkoitus olla sekä toimiva että helposti muokattava ja jatkokehitettävä.

Taideteosten ja mallien luonti

Pelaaja voi poimia käteensä ohuita, käytännössä kaksiulotteisia objekteja, jotka ovat muodoltaan vaihtelevia. Pelaaja voi halutessaan muuttaa objektin kokoa ja väriä sekä objektien rotaatiota. Halutun värin ja koon valinnan jälkeen pelaaja voi kiinnittää objektin kankaaseen lähellä olevan mallin mukaisesti. Kuten kuvassa 12 tarkoituksena on yksinkertaistaa maalauksen luomista niin, että sitä voivat tehdä kaikki taiteellisista kyvyistä riippumatta. Maalaukset luodaan mosaiikkimaisesti yksinkertaisista kappaleista, jotka ovat oikean kokoisia ja värisiä. Jos pelaaja kopioi maalauksen tarpeeksi hyvin, mosaiikki muutetaan pelissä näyttämään oikealta maalaukselta.

Toisissa tasoissa pelaaja voi poimia käteensä 3D-objekteja, jotka ovat myös muodoltaan vaihtelevia. Samanlailla kuin maalausten kanssa pelaaja voi vaihtaa objektien kokoa, väriä ja rotaatiota. Kun objekti on halutunlainen, se voidaan kiinnittää joko patsaan jalustaan tai toiseen objektiin, joka on kiinnitetty patsaan jalustaan.

Mallien rakentamisen tulee tapahtua Unity Editorissa. Mallia rakentaessa pelintekijä voi luoda uusia objekteja ja liimata ne kankaaseen tai jalustaan, jotta pelaaja voi yrittää kopioida mallin. Mallien tulee olla helposti rakennettavia, jotta uusia tasoja on helppo luoda.



Kuva 12. Suunnitelma maalauksen luonnista

Taideteosten vertailu malliin

Vertailun tulee tapahtua nopeasti ja luotettavasti. Maalausten vertailussa tulee ottaa huomioon vain näkyvä väri tietyssä kohtaa kangasta. Patsaiden vertailussa tulee ottaa huomioon objektien sijainti suhteessa jalustaan sekä objektien rotaatio, koko ja väri. Koska täsmälleen samojen arvojen saaminen tulee olemaan käytännössä mahdotonta, pitää vertailulle asettaa sopivat virhemarginaalit, joiden sisään pelaajan tulee saada objektit.

3.2 Haasteet ja alustavat ratkaisut

Ensimmäinen haaste oli sopivan taiteenluontijärjestelmän kehittäminen. Spraymaalusta muistuttavaa järjestelmää harkittiin, mutta se hylättiin liian vaikeana pelaajalle. Virtuaalitodellisuuslasien käsienseuranta toimii välillä heikonlaisesti, ja spraymaalauksen vaatisi myös enemmän taiteellisia kykyjä pelaajalta kuin prototyypin mosaiikkityyppinen maalausten rakentaminen.

Isoimpana haasteena oli sopivan vertailualgoritmin kehittäminen. Maalausten vertailuun ensimmäisenä suunnitelmana oli tallentaa kaikki kankaaseen kiinnitetyt objektit ja vertailla näitä malliin. Unityn lokaatiojärjestelmä teki tästä kuitenkin turhan hankalaa.

Unityssä jokaisella objektilla on lokaatioarvot kaikilla kolmella akselilla (x, y, z), ja tämä arvo otetaan objektin keskipisteestä. Mutta maalauksen tapauksessa objekti voisi sijaita missä vain arvoissa, mutta tämä ei vielä kertoisi mitään siitä, kuinka iso alueen objekti peittää. Tämän lisäksi projektin mosaiikkijärjestelmässä ideana oli se, että objektit voivat mennä päällekkäin, jolloin pelkästä lokaatiosta ei olisi hyötyä kertomaan, mikä alue objektista on näkyvissä ja mikä toisen objektin peittämä. Tästä suunnitelmasta siirryttiinkin nopeasti nyt käytössä olevaan tarkastuspistejärjestelmään.

Patsaiden vertailujärjestelmän alkuperäinen suunnitelma oli hyvin samanlainen kuin lopullinen versiokin. Alkuperäinen suunnitelma oli hiukan yksinkertaisempi, jossa kahdelta listalta otettiin samalla järjestysnumerolla olevat alkiot ja verrattiin näitä keskenään. Tämä suunnitelma ei kuitenkaan toiminut suoraan, koska pelaaja voi asettaa palikoita lähestulkoon missä järjestyksessä haluaa, joten lista ei olisi samassa järjestyksessä kuin mallipatsaan objekteista muodostettu lista. Näin syntyi suunnitelma kahdesta sisäkkäisestä silmukasta, jossa mallipatsaan listasta otetaan objekti, ja sitä verrataan kaikkiin pelaajan objekteihin. Sisäkkäiset silmukat ovat omiaan tuhoamaan pelin suorituskyvyn, mutta vain yhdellä tasolla sisäkkäin olevat silmukat, jotka ajetaan vain silloin, kun pelaaja niin haluaa, eivät juurikaan vaikuta pelin toimintaan normaalin pelaamisen aikana. Silmukat ajetaan vain silloin kun pelaaja manuaalisesti aloittaa vertailun, joten kyse ei ole usein tapahtuvasta operaatiosta. Tämän lisäksi patsaat eivät voi olla erityisen monimutkaisia sadoista tai tuhansista objekteista koostuvia jättiläisiä, koska se nostaisi vaikeustasoa ja yksittäisten tasojen kestoa todella paljon.

Koska insinööriyötä alettiin työstää jo työharjoittelun aikana, sen toteutus tapahtui käytännössä viikonloppuisin. Tämä aiheutti pieniä haasteita kehitystyössä, koska se tarkoitti sitä, että aikataulu oli rajallinen ja kehitykseen tuli pitkiä taukoja. Tämän takia olikin äärimmäisen tärkeää kartoittaa tarkkaan, minä päivänä tehtiin mitäkin. Työharjoittelun loputtua kehitys jatkui täyspäiväisesti projektin valmistumiseen saakka. Aikataulu ja suunnitelmat pitivät hyvin, työn kesto onnistuttiin arvioimaan oikein ja suurempia muutoksia tai yllätyksiä ei onneksi tullut.

4 Taiteenluontiprototyypin toteutus

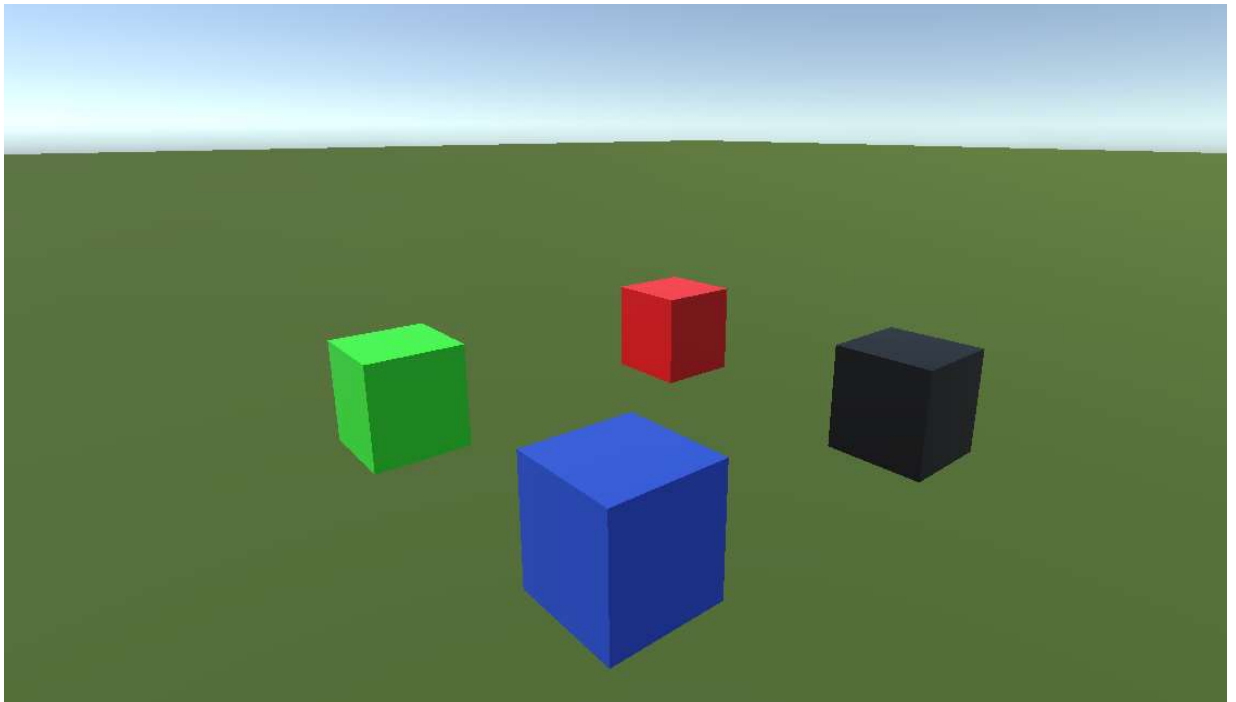
Koska kyseessä on prototyyppi, työssä ei juuri keskitytty ulkonäköseikkoihin vaan puhtaasti ominaisuuksien tarkoituksenmukaiseen toimintaan. Tästä syystä kaikki kappaleet ovat Unityn primitiivisiä 3D-objekteja eivätkä värit edusta lopullista värimaailmaa. Primitiiviset objektit voidaan jatkokehityksessä korvata varta vasten tehdyillä 3D-malleilla ja värimaailma vaihtaa sopivaksi. Kaikki ohjelmakoodi kehitettiin niin, että mallien muuttamisella ei pitäisi olla mitään vaikutusta ohjelman toimintaan.

Prototyyppi jakautuu selkeästi kahteen eri osaan, luomiseen ja vertailuun. Nämä osat jakautuvat edelleen kahteen osaan, maalauksiin ja patsaisiin. Tämä antoi työn toteutukselle selkeän rakenteen, tuottaa asiat yksi kerrallaan. Koska vertailua ei voi olla ilman luomista, luominen oli selkeä lähtökohta toteutukselle.

4.1 Maalausten ja patsaiden luonti

Maalaukset

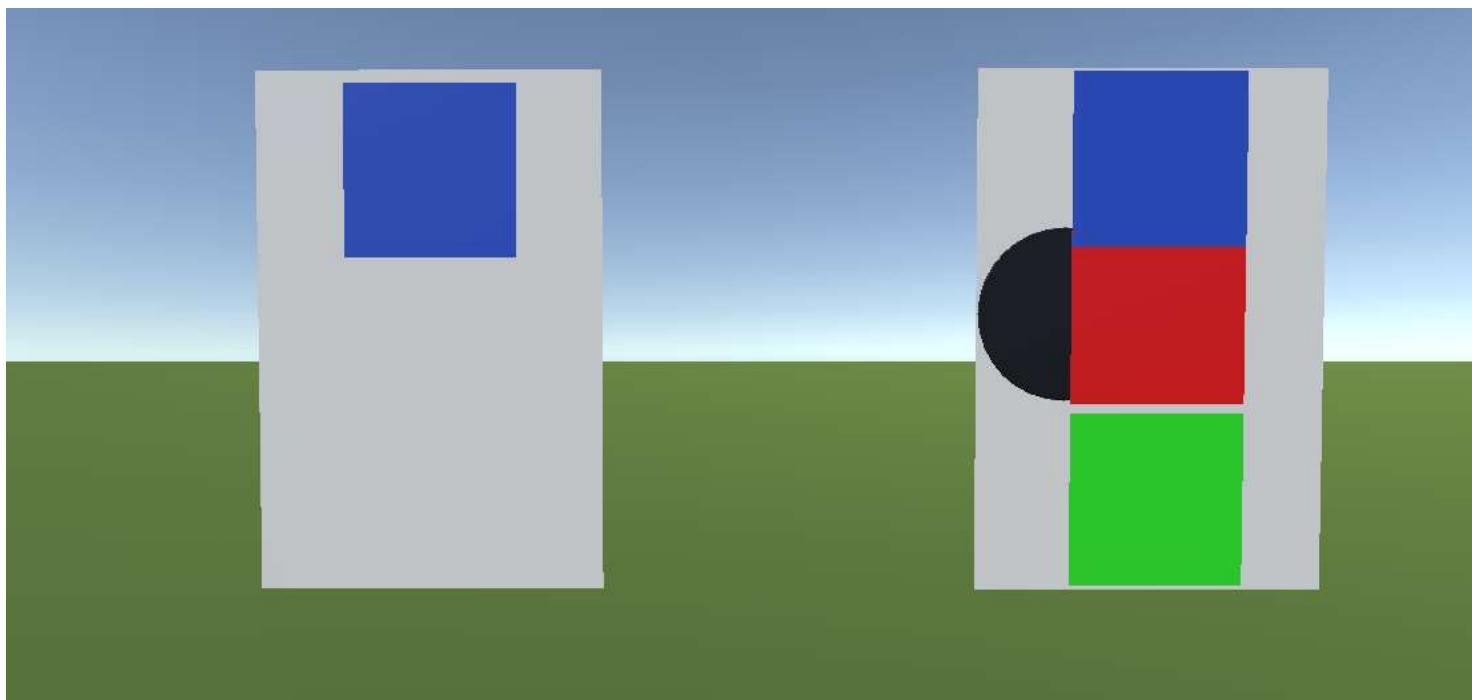
Kuten suunnitteluvaiheessa todettiin, varsinainen maalaus ja maalausten kopiointi ohjaimilla tulisi olemaan liian vaikeaa ja rajoittaisi paljon sitä, minkälaisia maalauksia pelissä voi olla. Tämän tilalle kehitettiin suunnitelma mosaiikkityylisestä ratkaisusta, jossa pelaajalla on eri muotoisia palasia joita ”liimata” kankaalle. Ennen liimausta pelaaja voi muuttaa palasen väriä ja kokoa. Prototyypin palaset tehtiin Unityn primitiivisistä 3D-objekteista, joista yksi ulottuvuus litistettiin hyvin pieneksi. Käytännössä pallo litistettiin ympyräksi ja kuutio neliöksi. Värin vaihto toteutettiin, kuvassa 13 näkyvillä, maaliämpäriä edustavilla värillisillä kuutioilla. Pelaaja ottaa palasen käteensä ja koskettaa sillä haluaomaansa maaliämpäriä, jolloin ämpäri kopioi oman värinsä palasen väriksi. Tämä saavutettiin käyttämällä Unityn oletusvarjostinta, joka kaikilla luoduilla objekteilla on valmiiksi. Pelin alussa maaliämpäri hakee oman väriarvonsa ja tallentaa sen muuttujaan. Törmäyksen tapahtuessa ämpäri hakee törmänneen objektin varjostin-komponentin ja muuttaa värin.



Kuva 13. "Maaliämpärit"

Objektin koon muutos tapahtuu ohjaimen liipaisinta ja ohjaussauvaa käyttämällä. Pelaaja painaa liipaisimen pohjaan ja tämän jälkeen kokoa voi muuttaa suuremmaksi ja pienemmäksi siirtämällä ohjaussauvaa joko ylöspäin koon kasvattamiseksi tai alaspäin koon pienentämiseksi. Liipaisimen pohjassa pitäminen lisättiin varotoimenpiteenä. Ohjaussauvat ovat hyvin herkäät, joten vahingossa tapahtuvia koon muutoksia tapahtuisi liikaa ilman "varmistusta".

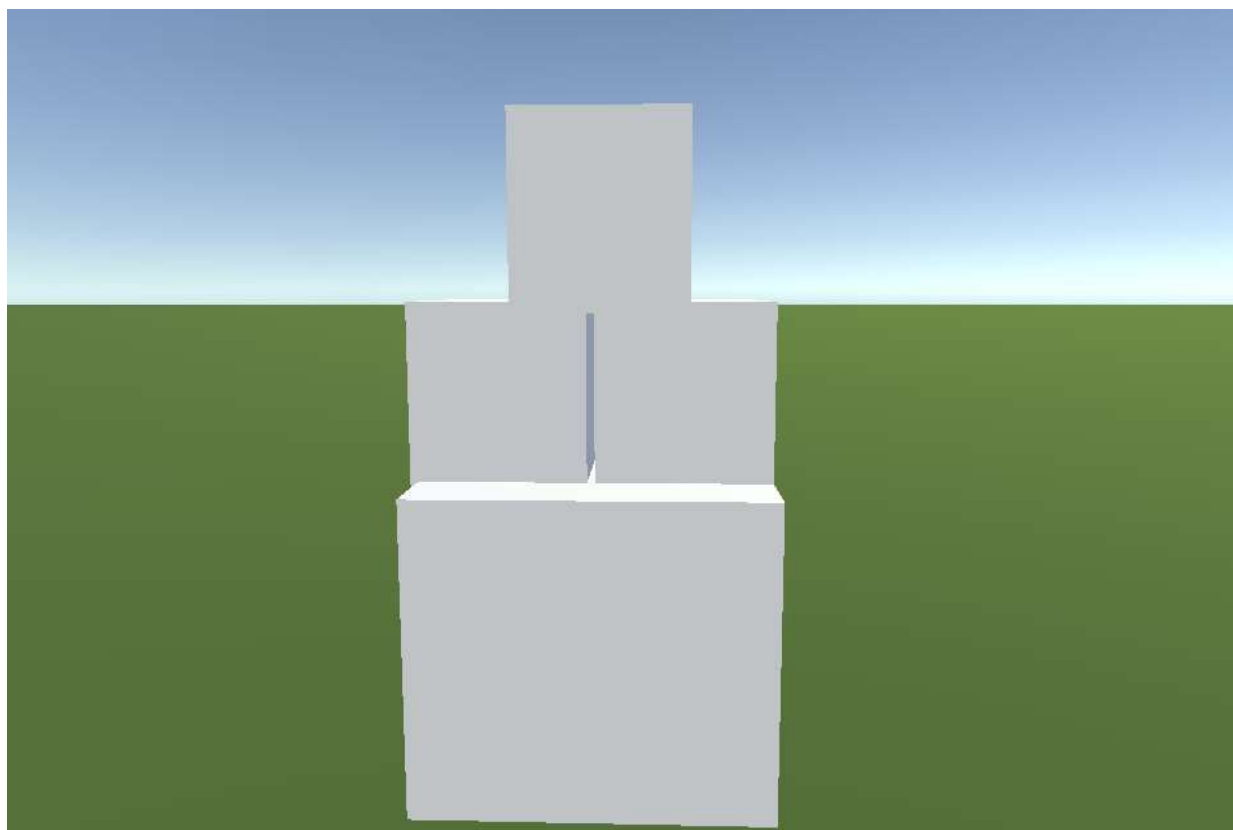
Itse liimaus tapahtuu suhteellisen yksikertaisesti. Pelaaja asettaa palasen haluamaansa kohtaan haluamallansa rotaatiolla ja painaa ohjaimen A-painiketta. Tämän jälkeen ohjelmakoodi tekee tarvittavat asiat, jotta palanen saadaan kiinnitettyä "kankaaseen". Kuvassa 14 nähdään pelaajan aloittama maalaus, jonka vieressä on mallikangas. Kun pelaaja poimii palasen, se asetetaan pelaajan käsiobjektin lapseksi. Kun palanen kiinnitetään kankaaseen, tämä vanhempi-lapsi-suhde poistetaan.



Kuva 14. Maalauksen luontia, malli oikealla

Patsaat

Patsaiden rakentaminen on pelaajan näkökulmasta samanlaista kuin maalaustenkin. Pelaaja voi vaihtaa palasten väriä ja kokoa samalla tavalla kuin maalauspalsastenkin. Erona on kuitenkin se, että maalauksissa pelaaja kiinnittää palasen aina kankaaseen. Patsaissa pelaaja voi kiinnittää palasen sekä jalustaan että toisiin palasiin, jotka ovat jo kiinni jalustassa. Prototyypissä patsaspalaset koostuivat kuutioista ja ympyröistä, mutta samanlailla kuin maalausten tapauksessa, jatkokehityksessä palaset voidaan korvata varta vasten tehdyillä 3D-malleilla. Kiinnityksessä palanen asetetaan jalustan lapsiobjektiksi ja lisätään jalustan listaan palasista vertailua varten.

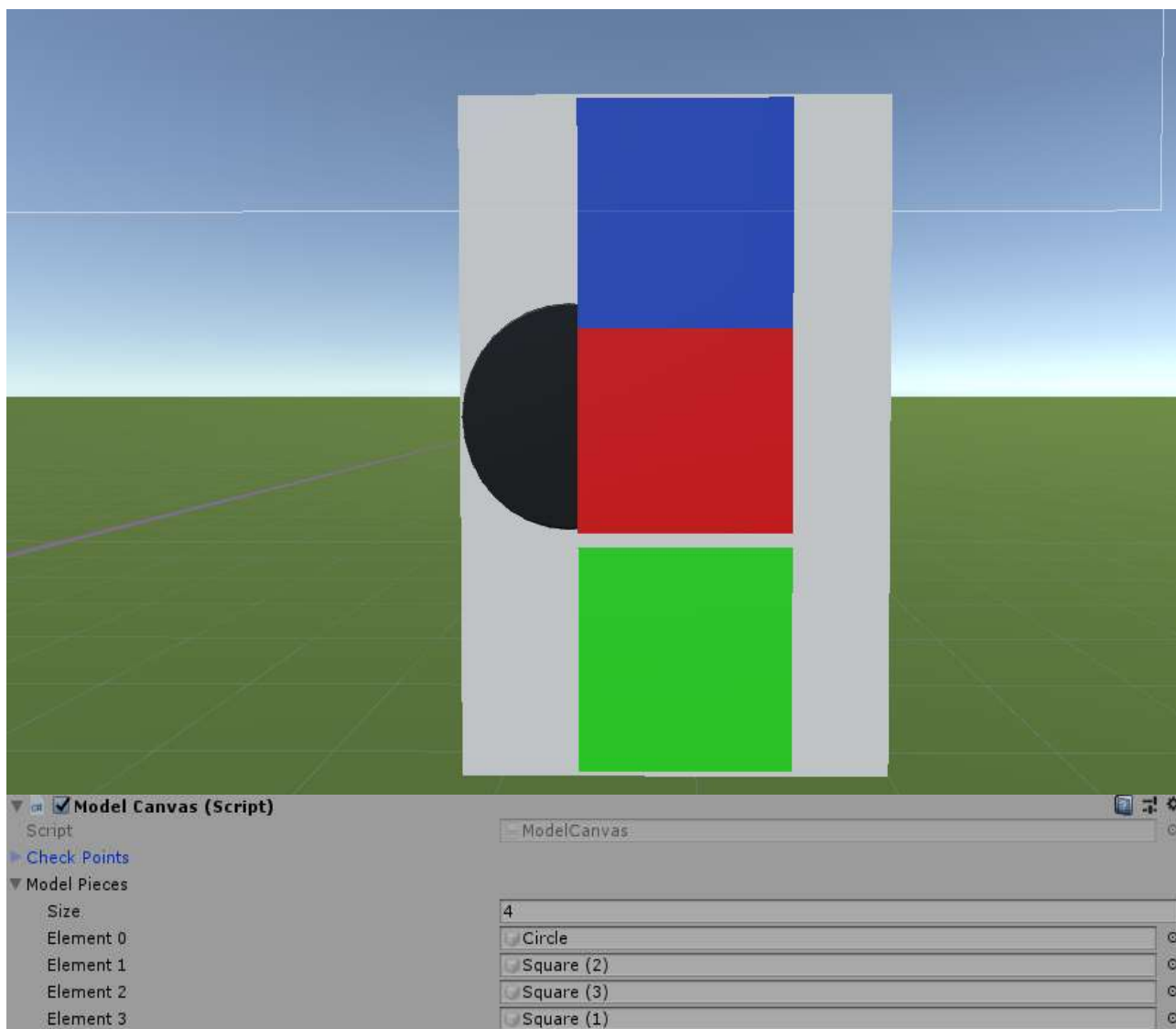


Kuva 15. Patsaan luontia, alin kuutio on jalusta ja kolme ylemmää ovat palikoita, joista pelaaja kasaa patsaan.

Kuvassa 15 on pelaajan rakentama patsas. Se koostuu jalustasta sekä kolmesta kuutiosta, jotka pelaaja on asettanut mallin mukaiseen järjestykseen.

4.2 Maalausten ja patsaiden vertailu malliin

Ennen kun voidaan vertailla pelaajan tuotosta, tarvitaan malli, johon verrata. Mallin rakentamista varten kehitettiin oma ohjelmakoodi. Pelintekijä voi Unity Editorissa asettaa mallin palaset haluamallaan tavalla ja tämän jälkeen hänen tarvitsee vain lisätä ne listaan, joka nähdään kuvassa 16. Pelin alkaessa tämä lista käydään läpi ja palaset kiinnitetään kankaaseen, jalustaan tai toisiinsa tilanteen mukaan. Malli on sen jälkeen valmis, ja pelaaja voi alkaa kopioimaan sitä.



Kuva 16. Valmis malli. Alla näkyy lista, johon mallin palaset lisätään.

Toisin kuin maalausten ja patsaiden luontikoodit vertailukoodit eroavat toisistaan huomattavasti. Tämä johtuu lähinnä siitä, että maalaukset ovat kaksiulotteisia ja patsaat kolmiulotteisia.

Maalaukset

Maalausten vertailu perustuu värien tarkasteluun. Koska viimeiseksi asetettu palanen jää aina päällimmäiseksi, voidaan vain tarkastella päällimmäisenä olevia värejä eri kohdissa.

Prototyypissä kankaalla on 110 tarkastelupistettä asetettuna 10 * 11 -taulukkoon. Tämä johtuu siitä, että kangas ei ole neliö, vaan hieman leveyttään korkeampi. Nämä tarkastelupisteet tallennetaan listaan. Kun pelaaja ilmoittaa olevansa valmis, jokaisesta pisteestä ammutaan säde kohti kangasta ja tallennetaan sen objektin väriarvo, johon osuttiin. Tämä tehdään sekä pelaajan kankaalle että mallikankaalle.



Kuva 17. Pelaajan tekemää maalausta verrataan oikealla olevaan mallikappaleeseen.

Tämän jälkeen tallennetut osumat käydään läpi yksitellen. Otetaan sekä pelaajan kankaan listasta, että mallikankaan listasta ensimmäisen tarkistuspisteen osumat. Näiden objektien väriarvo jaetaan kolmeen liukulukuun (RGB-arvot) ja niitä verrataan toisiinsa: punaisen arvo punaiseen ja niin edelleen. Mikäli kaikki kolme arvoa ovat virhemarginaalin sisällä, lisätään tarkistuspiste onnistuneiden listaan. Lopuksi selvitetään onnistumisprosentti jakamalla onnistuneiden pisteiden määrä kaikkien pisteiden määrällä ja kerrotaan tulos sadalla. Tämä tulos näytetään pelaajalle, kuten kuvassa 17. Vaikeustasoa

voidaan nostaa pienentämällä virhemarginaalia, lisäämällä tarkastuspisteiden määrää tai vaatimalla korkeampaa onnistumisprosenttia tason läpäisyyn. Vastaavasti virhemarginaalin kasvatus, tarkastuspisteiden määrän laskeminen ja vaaditun onnistumisprosentin laskeminen helpottaa peliä.

Tässä prototyypissä tarjolla olevien värien määrä on rajallinen (musta, punainen, sininen ja vihreä), mutta jatkokehityksessä pelaaja voi mahdollisesti sekoittaa omat värinsä ja tämä tarkistusjärjestelmä toimisi hyvin edelleen. Omien värien sekoitus lisäisi virhemarginaalin merkitystä tarkastuksessa huomattavasti.

```
modelR = modelAnim.GetColor().r;
modelG = modelAnim.GetColor().g;
modelB = modelAnim.GetColor().b;

playerR = playerAnim.GetColor().r;
playerG = playerAnim.GetColor().g;
playerB = playerAnim.GetColor().b;

if (playerR > modelR - 0.01 && playerR < modelR + 0.01)
{
    if (playerG > modelG - 0.01 && playerG < modelG + 0.01)
    {
        if (playerB > modelB - 0.01 && playerB < modelB + 0.01)
        {
            return true;
        }
    }
}
```

Esimerkkikoodi 1. Koodi hakee molempien objektien väriarvot (R, G, B) ja vertailee niitä keskenään.

Patsaat

Patsaiden vertailu on huomattavasti vaativampaa, koska pelkkä väriarvojen tarkastelu ei tietenkään riitä. Vertailu ei myöskään käytä ulkoisia tarkastuspisteitä vaan vertaa objekteja suoraan toisiinsa.

Mallijalusta sekä pelaajan jalusta sisältävät listat kaikista kiinnitetyistä objekteista. Objektit ovat myös jalustan lapsia, joten vertailussa voidaan käyttää lokaaleja arvoja globaalien sijaan. Unityssä globaalit arvot tietyille asioille, kuten sijainnille, ovat arvoja, jotka ovat suhteessa koko maailman nollakohtaan. Lokaalit arvot puolestaan ovat arvoja, jotka ovat suhteessa objektin vanhempaan [23]. Esimerkiksi jos yksi objekti on kaksi etäisyysyksikköä kohtisuoraan pelaajan jalustan yläpuolella ja toinen kaksi etäisyysyksikköä kohtisuoraan mallijalustan yläpuolella, ainoastaan objektien korkeussuunnan (y) globaali

arvo olisi sama ja kaksi muuta (x, z) eri. Mutta lokaalit arvot olisivat kaikki täsmälleen samat, koska objektien sijainti suhteessa omaan vanhempaan on täsmälleen sama. Kun pelaaja on saanut patsaansa rakennettua ja painaa vertailunappia, vertailu alkaa. Koodi ottaa listat sekä mallijalustalta että pelaajan jalustalta ja alkaa käymään niitä läpi sisäkkäisissä for-silmukoissa. Ulompi for-silmukka käy läpi jokaisen mallijalustaan kiinnitetyn objektin, yksi kerrallaan. Sisempi for-silmukka vertaa jokaista objektia kaikkiin objekteihin pelaajan jalustassa.

```
for (int i = 0; i < modelObjectsCount; i++)
{
    model = modelObjects[i].transform;

    playerObjectsCount = playerObjects.Count;
    for (int j = 0; j < playerObjectsCount; j++)
    {
        player = playerObjects[j].transform;

        if (CompareLocation(model, player))
        {
            if (Quaternion.Angle(model.rotation, player.rotation) < 5)
            {
                if (CompareScale(model, player))
                {
                    if (CompareColor(model.gameObject, player.gameOb-
ject))
                    {
                        correctObjects += 1;
                    }
                }
            }
        }
    }
}
```

Esimerkkikoodi 2. Koodi vertailee mallipatsaan osia pelaajan rakentaman patsaan osiin sisäkkäisissä for-silmukoissa.

Vertailun kohteena ovat lokaali sijainti, rotaatio, skaala (objektin koko) sekä objektin väri. Sijainnin vertailu käyttää CompareLocation-metodia, joka hakee molempien objektien lokaalit sijaintiarvot ja vertailee niitä toisiinsa. Rotaation vertailuun puolestaan käytetään Unityn valmista metodia Quaternion.Angle, joka vertailee kahden objektin välistä kulmaa. Rotaation saaminen oikein valkoisilla kuutioilla on suhteellisen haastavaa, koska ne näyttävät samalta käänsi kuutioita miten päin tahansa. Lopullisessa pelissä kuitenkin objektit tulevat olemaan vaihtelevampia, tai vähintäänkin niillä on oma väritys, joten ongelma poistuu. Skaalaa, eli kokoa vertaillaan samalla tavalla kuin sijaintiakin. Erillinen

metodi hakee objektien skaala-arvot ja vertaa niitä toisiinsa virhemarginaalin kanssa. Värivertailuun käytetään samaa koodia kuin maalausten vertailussa. Samanlailla kuin maalausten tapauksessa, mikäli mallijalustan objektille löytyy vastine pelaajan objekteista, nostetaan onnistuneitten määrää. Kaikilla arvoilla on myös pieni virhemarginaali, koska millintarkka objektien sijoittaminen on täysin mahdotonta. Tässäkin vaikeusastetta voi nostaa tai laskea virhemarginaalia säätämällä. Toisin kuin maalauksissa, vaikeusaste patsaissa riippuu käytännössä kokonaan virhemarginaalista. Lopuksi arvo näytetään pelaajalle, kuten kuvassa 18.



Kuva 18. Pelaajan tekemää patsasta verrataan oikealla olevaan mallikappaleeseen

Koska valmiiksi määritellyt tarkistuspisteitä ei ole, yhden palasen prosentuaalinen arvo riippuu siitä, kuinka monta palasta valmiissa patsaassa on. Tätä eroa voidaan jatkokehityksessä tasoittaa esimerkiksi useammalla virhemarginaalitasolla. Esimerkiksi täsmälleen sama tulos antaisi enemmän pisteitä kuin 0,2 yksikön heitto, mutta pienellä heitollakin saisi pisteitä eikä suoraa hylkäystä.

4.3 Ongelmat ja ratkaisut

Projekti saatiin valmiiksi ilman suurempia ongelmia toteutuksessa. Isoimmat ongelmat olivat tarkistuksen toteutus sekä sopivien kontrollien löytäminen.

Tarkistuksessa käydään useampia listoja läpi, joten suorituskyky muodostui isoksi huoleksi. Loppujen lopuksi tämä huoli oli kuitenkin aiheeton. Vertailuja tehdään hyvin harvoin, ja silloinkin vertailtavia objekteja ei ole enempää kuin korkeimmillaan joitain satoja. Tarkistuksen tarkkuus, ennen kaikkea maalauksissa, oli myös huolenaihe. Tarkistuspisteet käyttävät Unityn fysiikkasäteitä (RayCast), jotka toisinaan saattavat toimia epäluotettavasti. Tarkistuspisteet ovat kuitenkin niin lähellä kangasta, että niiden toiminta oli varmaa. Varasuunnitelmana olisi ollut käyttää samaa tapaa kuin patsaiden kanssa, mutta se olisi lisännyt turhaa monimutkaisuutta koodiin ja nykyinen järjestelmä toimii hyvin.

Kontrollien kanssa ongelmaksi muodostui lähinnä oikeiden yhdistelmien löytäminen sekä ”vahinkolaukausten” minimointi. Peli on kehitetty Oculus Rift S:llä, ja Oculuksen Touch-ohjaimet ovat todella herkäät. Tämä on hyvä koska se tarkoittaa, että pelaajan ei tarvitse kamppailla ohjaimen kanssa, mutta tarkoittaa myös sitä, että vahingossa tapahtuvia painalluksia tapahtuu kaikille. Tämän lisäksi ohjainten seuraamisessa on välillä parantamisen varaa.

Vahinkojen minimoimiseksi käytettiin selkeää varmistuskontrollia. Pelaajan täytyy painaa liipaisin pohjaan ennen kuin palikan kokoa voi muuttaa ja painaa A-näppäintä ennen kuin palikka kiinnittyy mihinkään. Tämä estää vahingossa tapahtuvien sauvaohjaimen liikkeiden sekä vahingossa tapahtuvien törmäysten vaikuttamisen peliin ja parantaa pelikokemusta.

5 Tulokset

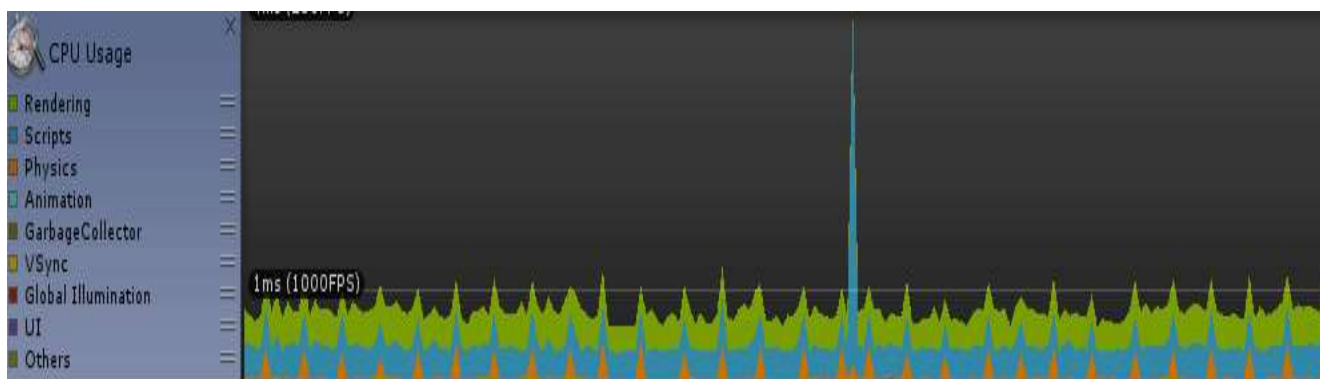
Insinööriyön suunnittelu alkoi keväällä 2019 ja varsinainen toteutus tapahtui samana kesänä. Suunnitelma ja työn laajuus pysyivät samana koko projektin ajan, mikä helpotti toteutusta.

5.1 Projektin saavutukset ja analyysi

Työn tavoitteena oli tuottaa toimiva prototyyppi, jota voidaan myöhemmin jatkokehittää valmiiksi peliksi ja tässä onnistuttiin. Prototyyppi sisältää kaikki vaaditut ominaisuudet ja koodi on suunniteltu jatkokehitys huomioon ottaen. Kappaleiden koon muuttaminen onnistuu kappaleen muodosta riippumatta, samoin kiinnitys.

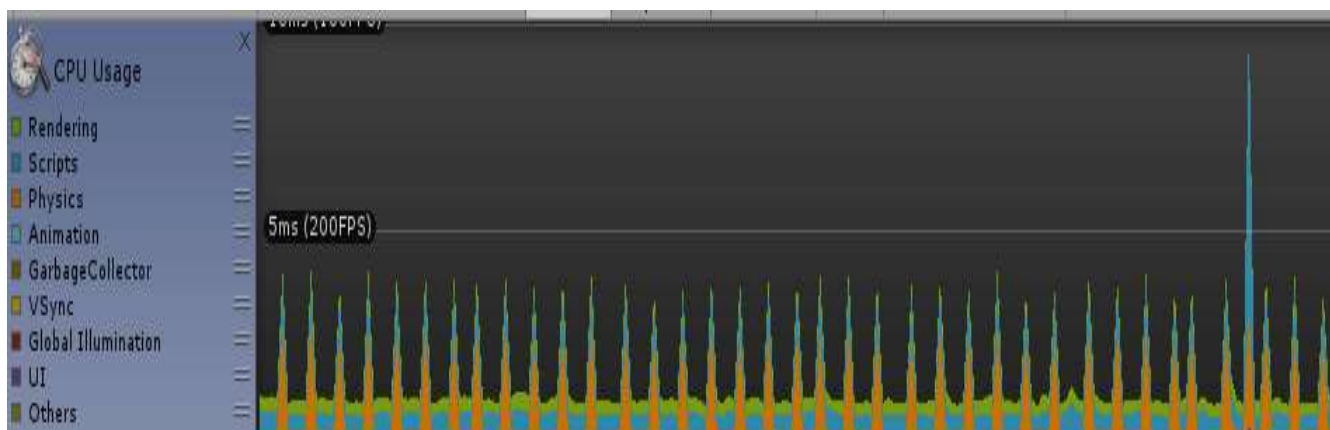
Patsaiden vertailu ei välitä kappaleiden muodosta vaan ainoastaan vertailee sitä mallikappaleen sijaintiin, rotaatioon, kokoon ja väriin. Maalausten vertailussa on otettu huomioon tarkistuspisteiden määrän vaihtelu ja annettu mahdollisuus toteuttaa pelaajan omien värien sekoittaminen. Eri vaikeustasojen luominen on mahdollistettu antamalla vertailulle muokattavat virhemarginaalit.

Alussa huolenaiheena ollut vertailun suorituskykyään ei lopulta muodostunut ongelmaksi. Kuvassa 19 nähdään selkeästi vertailusta aiheutuva piikki suorituskyvyssä. Prosessorin kuormitus luonnollisesti nousee vertailun aikana, mutta ei missään vaiheessa nouse ongelmaksi. Kuvassa koodin osuus on merkattu sinisellä ja fysiikan osuus oranssilla. Vihreä kuvaa grafiikoiden piirtoa.



Kuva 19. Unityn profiloija vertailutilanteessa

Piikin kohdalla ajetaan sekä maalauksen vertailu 110 tarkistuspisteellä sekä patsaan vertailu, jossa sekä mallipatsas että pelaajan patsas koostuvat 10 palasesta. Profiloijassa suorituskky ei ikinä laske alle 500 ruudunpäivityksen, joten ongelmia ei selkeästi ole pienemmillä patsailla.



Kuva 20. Unityn profiloija suuremmassa vertailutilanteessa

Kuvassa 20 nähdään selkeästi suurempi piikki. Siinä ajetaan sekä maalauksen vertailu 110 tarkistuspisteellä että patsaan vertailu, jossa molemmat patsaat koostuvat 96 palasesta. Profiloijassa näkyy selkeästi palasten määrän nousu lisääntyneenä fysiikan laskuna. Vertailun aiheuttama piikki näkyy edelleen selkeästi, mutta niinkin äärimmäisen suurella patsaalla kuin 96 palaa, suorituskky pysyy siedettävänä, yli 100 ruudunpäivitystä sekunnissa. Näin suuria patsaita tuskin kuitenkaan varsinaisessa pelissä tulee olemaan ja tämä olikin vain räsitustesti, jolla voitiin todeta suorituskvyn pysyvän hyvänä myös ääriolosuhteissa.

5.2 Jatkokehitys valmiiksi tuotteeksi

Jatkokehityksen kannalta sekä taideteosten luominen että vertailu ovat valmiita ominaisuuksia. Unityn primitiivisten kappaleiden korvaaminen tarkoituksenmukaisilla 3D-objek-

teilla, sekä tasojen ympäristön luominen ovat tärkeimmät lisäykset näihin ominaisuuksiin. Pelaajalle voidaan antaa mahdollisuus sekoittaa omat värinsä ja vaikeustasoja voidaan lisätä säätämällä virhemarginaaleja.

Valmiiseen peliin on tarkoitus tulla vielä kolmas ominaisuus, hiippailu museossa. Tämän ominaisuuden tarkoituksena on pistää pelaaja taidevarkaan saappaisiin. Pelaaja hiippaillee taidemuseoon taidejäljennöksensä kanssa ja varastaa alkuperäisen, ja jättää jäljennöksen tilalle. Tämän ympärille voidaan rakentaa pelin ydin. Pelaaja kopioi taideteoksen mahdollisimman tarkasti ja käy vaihtamassa sen oikean tilalle. Tästä pelaaja palkitaan rahalla tai vastaavalla kerättävällä asialla. Seuraavassa tasossa sama asia uudestaan. Tarinalla voidaan luoda syyt sille, miksi pelaajan on kerättävä rahaa tällä tavalla.

Tarinamuotoisen kampanjan lisäksi valmiiseen peliin voidaan luoda muitakin pelimuotoja. Esimerkiksi moninpelimuoto, jossa useampi pelaaja yrittää kopioida saman taideteoksen vuorotellen. Tämä voidaan toteuttaa myös samanaikaisena moninpelinä internetin kautta. Peliin voidaan lisätä myös päivittäisiä tai viikoittaisia haastetehtäviä.

6 Yhteenveto

Insinööriyössä suunniteltiin ja kehitettiin prototyyppi taiteen luomista ja vertailua varten virtuaalitodellisuusympäristössä. Työhön kuului prototyypin suunnittelu ja toteutus. Työn aikana tutustuttiin erilaisiin tapoihin toteuttaa sekä luomisprosessi että vertailualgoritmi.

Projekti eteni suunnitelman mukaisesti ja saatiin päätökseen suunnitellussa ajassa. Suunnitelma myös pysyi suurin piirtein saman läpi projektin, joka omalta osaltaan helpotti asioita. Muutamia ongelmia toteutuksen aikana tuli, mutta kaikki saatiin selvitettyä.

Suurimmat ongelmat olivat kontrollit sekä oikeanlaisen vertailualgoritmin kehittäminen. Molemmat kuitenkin saatiin ratkaistua tyydyttävällä tavalla. Kontrollit ovat sekä toimivat että myöhemmin helposti muokattavat. Vertailualgoritmi toimii luotettavasti ja riittävän tehokkaasti.

Työn tuloksena oli prototyyppi, joka täytti asetetut vaatimukset ja on valmis jatkokehitykseen. Taideteosten luominen ja vertailu onnistuivat pelimekaniikkoina ja ne voidaan lisätä tulevaan peliin lähestulkoon sellaisenaan. Molemmat pelimekaniikat ovat helposti säädettävissä ja jatkokehitettävissä.

Jatkokehitys tapahtunee myöhemmässä vaiheessa, Critical Charm Oy:n edellisen projektin valmistuttua. Prototyyppiä voidaan käyttää suoraan projektissa ja sen ympärille on helppo rakentaa kokonainen peli.

Lähteet

- 1 Kielitoimiston sanakirja. 2018. Verkkoaineisto. Kotimaisten kielten keskus. <www.kielitoimistonsanakirja.fi/virtuaalitodellisuus>. Luettu 9.9.2019.
- 2 Brockwell, Holly. 2016. Forgotten genius: the man who made a working VR machine in 1957. Verkkoaineisto. <www.techradar.com/news/wearables/forgotten-genius-the-man-who-made-a-working-vr-machine-in-1957-1318253>. Luettu 9.9.2019.
- 3 Google Patents. Verkkoaineisto. Google. <patents.google.com/patent/US3050870A>. Luettu 9.9.2019.
- 4 Google Patents. Verkkoaineisto. Google. <patents.google.com/patent/US2955156A>. Luettu 9.9.2019.
- 5 Sutherland, Ivan E. 1968. A head-mounted three dimensional display. Fall Joint Computer Conference. s.757-764.
- 6 Kaur, Gaganpreet. Virtual Reality. Verkkoaineisto. <medium.com/@kaurgagan073/virtual-reality-5a0164a2a3c2>. Luettu 10.9.2019.
- 7 Hämäläinen, Tuukka. 2016. Virtuaalitodellisuus tulee taas: Katsaus VR-laitteiden hulluun historiaan. Verkkoaineisto. <muropaketti.com/tietotekniikka/virtuaalitodellisuus-tulee-taas-katsaus-vr-laitteiden-hulluun-historiaan/>. Luettu 10.9.2019.
- 8 Engler, Craig E. 1992. Affordable VR by 1994. Computer Gaming World 1.11.1992, s.80-82.
- 9 Hill, Matt. 2014. The Sega VR Headset That Never Was. Verkkoaineisto. <www.gizmodo.co.uk/2014/11/the-sega-vr-headset-that-never-was>. Luettu 10.9.2019.
- 10 Seibert, William. 2017. Virtual Reality Then: A Look Back at the Nintendo Virtual Boy. <www.techspot.com/article/1085-nintendo-virtual-boy>. Luettu 10.9.2019.
- 11 Consolidated Sales Transition by Region. Verkkoaineisto. Nintendo. <www.nintendo.co.jp/ir/library/historical_data/pdf/consolidated_sales_e1509.pdf>. Luettu 10.9.2019.
- 12 Virtual Boy. Verkkoaineisto. Wikipedia. <en.wikipedia.org/wiki/Virtual_Boy>. Luettu 10.9.2019.
- 13 Glasstron. Verkkoaineisto. Wikipedia. <en.wikipedia.org/wiki/Glasstron>. Luettu 10.9.2019.

- 14 Kumparak, Greg. 2014. A Brief History of Oculus. Verkkoaineisto. <techcrunch.com/2014/03/26/a-brief-history-of-oculus>. Luettu 10.09.2019.
- 15 Do, Tuan. 2018. List of Smartphones Powered by Qualcomm Snapdragon 835 Processor. Verkkoaineisto. <www.techwalls.com/qualcomm-snapdragon-835-smartphones>. Luettu 10.09.2019.
- 16 Delfino, Devon. 2019. 'What is Google Cardboard?': Everything you need to know about Google's low-tech VR experience. Verkkoaineisto. <www.businessinsider.com/what-is-google-cardboard>. Luettu 10.09.2019.
- 17 13 Of The Best Virtual Reality Uses In Industry To Date. 2019. Forbes. <www.forbes.com/sites/forbestechcouncil/2019/01/23/13-of-the-best-virtual-reality-uses-in-industry-to-date/#458994ba220>. Luettu 10.9.2019.
- 18 Skirka, Hayley. 2019. The reality of virtual travel: could it replace actual holidays?. Verkkoaineisto. <www.thenational.ae/lifestyle/travel/the-reality-of-virtual-travel-could-it-replace-actual-holidays-1.822346>. Luettu 10.9.2019.
- 19 Sablich, Justin. 2019. The Future Is Here, Almost: Virtual Travel Becomes More Of A Reality. Verkkoaineisto. <www.nytimes.com/2019/02/22/travel/virtual-reality-airlines-restaurants-travel.html>. Luettu 10.09.2019.
- 20 Tilt Brush. Verkkoaineisto. <store.steampowered.com/app/327140/Tilt_Brush>. Luettu 10.9.2019.
- 21 Matney, Lucas. 2018. Facebook brings animation tools to its Quill VR painting app. Verkkoaineisto. <techcrunch.com/2018/02/08/facebook-brings-animation-tools-to-its-quill-vr-painting-app/>. Luettu 10.09.2019.
- 22 Barker, Sammy. 2018. CoolPaint VR Review. Verkkoaineisto. <www.pushsquare.com/reviews/ps4/coolpaintrvr>. Luettu 10.9.2019.
- 23 Transform.localPosition. Verkkoaineisto. <docs.unity3d.com/ScriptReference/Transform-localePosition.html>. Luettu 13.9.2019.